

COMPUTER NETWORKS

ECHO PROGRAM

SERVER PROGRAM

```
#include <stdio.h>

#include <stdlib.h>

#include <unistd.h>

#include <arpa/inet.h>

#include <string.h>


int main(int argc, char const *argv[]) {

    int serverFd, clientFd; struct
sockaddr_in server, client; int
len; int port = 1234; char
buffer[1024]; if (argc == 2) {
port = atoi(argv[1]);
}

serverFd = socket(AF_INET, SOCK_STREAM, 0);

if (serverFd < 0) {

    perror("Cannot create socket");

    exit(1);
```

```

}

server.sin_family = AF_INET; server.sin_addr.s_addr =
INADDR_ANY; server.sin_port = htons(port); len =
sizeof(server); if (bind(serverFd, (struct sockaddr
*)&server, len) < 0) { perror("Cannot bind sokcet");
    exit(2);
}

if (listen(serverFd, 10) < 0) {
perror("Listen error"); exit(3);
}

while (1) { len = sizeof(client); printf("waiting for clients\n"); if
((clientFd = accept(serverFd, (struct sockaddr *)&client, &len)) < 0) {
perror("accept error");
    exit(4);
}

    char *client_ip = inet_ntoa(client.sin_addr); printf("Accepted new connection from a
client %s:%d\n", client_ip, ntohs(client.sin_port)); memset(buffer, 0, sizeof(buffer));

    int size = read(clientFd, buffer, sizeof(buffer));

    if ( size < 0 ) {
perror("read error");
exit(5);
    }

    printf("received %s from client\n", buffer);

    if (write(clientFd, buffer, size) < 0) {
perror("write error"); exit(6);

```

```
    }  
    close(clientFd);  
}  
close(serverFd);  
return 0;  
}
```

CLIENT PROGRAM

```
#include <stdio.h>  
  
#include <stdlib.h>  
  
#include <unistd.h>  
  
#include <arpa/inet.h>  
  
#include <string.h>  
  
  
const char message[] = "Hello sockets world\n";  
  
  
int main(int argc, char const *argv[]) {  
  
    int serverFd; struct  
    sockaddr_in server;
```

```
int len; int port = 1234;

char *server_ip = "127.0.0.1";

char *buffer = "hello server";

if (argc == 3) {

    port = atoi(argv[2]);

}

serverFd = socket(AF_INET, SOCK_STREAM, 0);

if (serverFd < 0) {

perror("Cannot create socket");

    exit(1);

}

server.sin_family = AF_INET;

server.sin_addr.s_addr = inet_addr(server_ip);

server.sin_port = htons(port);

len = sizeof(server); if (connect(serverFd, (struct sockaddr

*)&server, len) < 0) {    perror("Cannot connect to server");

    exit(2);

}

if (write(serverFd, buffer, strlen(buffer)) < 0) {

perror("Cannot write");

    exit(3);

}
```

```

char recv[1024]; memset(recv, 0,
sizeof(recv)); if (read(serverFd, recv,
sizeof(recv)) < 0) { perror("cannot
read");

    exit(4);

}

printf("Received %s from server\n", recv);

close(serverFd); return 0;

}

```

